

# 基于 Anime 的 Stable Diffusion 画风迁移微调

中国人民大学 李修羽

## 1 题目选择

Stable Diffusion 微调的目标，是将新概念注入预训练模型，利用新注入的概念以及模型的先验知识，基于文本引导条件生成自定义图片。目前主流训练 Stable Diffusion 模型的方法有 Full FineTune、Dreambooth、Text Inversion 和 LoRA。

一般的微调使用的数据集都是风格强烈鲜明的图片（例如油画、剪纸、国风）等，易于训练。基础的 Stable Diffusion 模型（sdv1-5, sdxl 等）也在此类图片上适配性较好。对于偏卡通、动漫人物（Anime）的绘图，基础模型或全量微调后的模型对 prompt 的要求高，普通微调的效果并不好。

笔者基于来自同一画风的两个 Anime 人物构建数据集（50 images），尝试画风 LoRA 和人物 LoRA 等结合，实现让 sdxl 模型能基于 prompt 绘制出从未学习过的人物，并达成高完成度、美观性、画风一致性。

## 2 数据集处理

由于 Anime 绘图对 prompt 的要求高，如果在 token 里仅描述部分特征，训练出来的更偏向为画风 LoRA，容易出现过拟合、模糊、人物崩坏等情况。对于更精确的人物 LoRA，笔者采用了 Diffusion Tagger<sup>1</sup> 进行 prompt 的处理，提高 character tag 的权重，并进行清理，得到适合 Amine 图片的 text 描述。

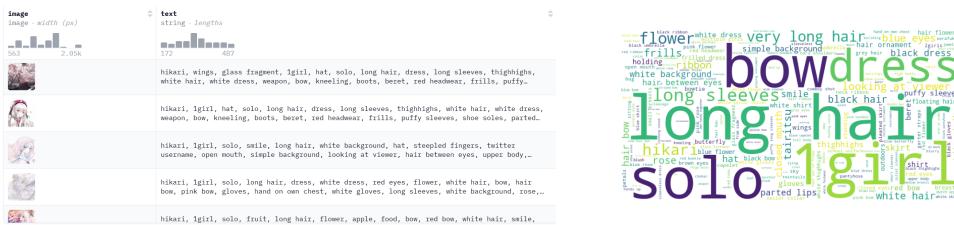


图 1: 数据集与 token 词云图

<sup>1</sup><https://huggingface.co/SmilingWolf/wd-swinv2-tagger-v3>

完成后的数据集<sup>2</sup>图片大小平均在  $1024 \times 1024$  以上，适合高分辨率训练，token 数也提升到平均约 30 条。



图 2: 画风迁移对象

### 3 训练

对基于 sdxl-1.0 全量微调的 animagine-xl<sup>3</sup> 模型进行 LoRA 微调，分辨率为 1024，使用 fp16 进行训练，训练时长  $24 \text{ RTX4090} \cdot h$ 。

画风 LoRA 训练图片平均 token 数为 8，人物 LoRA 训练图片平均 token 数为 30。由于 token 数较少，画风 LoRA 的训练结果出现过拟合导致的图片崩坏，采用前期的 checkpoint 作为权重。人物 LoRA 训练时收敛良好。

### 4 推理

#### 4.1 推理流程

使用 `DiffusionPipeline`，尝试使用训练的画风 LoRA、人物 LoRA、以及加速推理的 LCM-LORA<sup>4</sup> 作为权重进行推理。

<sup>2</sup><https://huggingface.co/datasets/sheriyuo/arcae>

<sup>3</sup><https://huggingface.co/Linaqruf/animagine-xl>

<sup>4</sup>Luo S, Tan Y, Huang L, et al. Latent consistency models: Synthesizing high-resolution images with few-step inference[J]. arXiv preprint arXiv:2310.04378, 2023.

## 4.2 图片生成结果

由图 3 可得，纯人物 LoRA 的绘图一致性远优于其余 LoRA 权重，LCM-LoRA 权重也会影响图片生成的对比度、相似度、画风。



图 3: "face focus, cute, masterpiece, best quality, arcaeae, hikari, solo, long hair, dress, white dress, flower, bow, white hair, looking at viewer, hat, long sleeves, very long hair, hair bow, rose, frilled dress, ribbon, outdoors, day, glass fragment"

## 4.3 推理加速

LCM-LoRA 对于推理的的画风影响较大，无法直接通过 adapter 层面进行调整，需要对 LCM-LoRA 进行单独的微调，并对 CFG 参数进行调整。

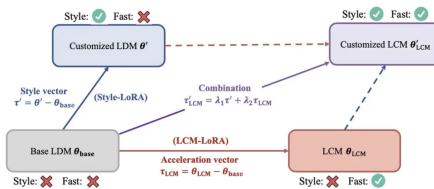


图 4: Overview of LCM-LoRA

如果只调整 `adapter_weights`, 降低 LCM-LoRA 的权重可以减小其影响, 但也会导致推理步数的增加, 从而降低加速效果。同时, 引入 `LCMScheduler` 导致的收敛问题会使得 LCM-LoRA 权重较小时推理结果出现白雾模糊, 无法移除。

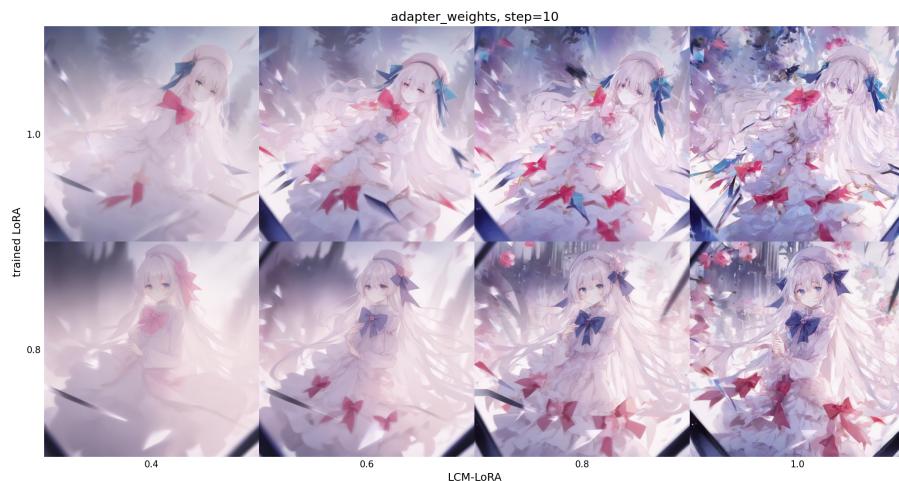


图 5: 权重调整